

A reconfigurable architecture for the FFT operator in a Software Radio context

Ali AL GHOUWAYEL, Yves LOUËT and Jacques PALICOT
Supélec-IETR, Avenue de la Boulaie BP 81127
35511 CESSON-SEVIGNE Cedex, France
Email: {ali.alghouwayel, yves.louet, jacques.palicot}@supelec.fr

Abstract—The "SoftWare Radio (SWR)" concept has become a topic of widespread interest for reconfigurable mobile architecture design. It is seen as the next evolutionary step in the mobile communications. In this context of SWR, a way to decrease the runtime of the software reconfiguration and to optimize the sharing between the software and the hardware of the execution platform called "parametrization" was introduced. This technique is based on two approaches, the first one is called the Common Function approach, the second one is called the Common Operator approach. Being interested on the second parametrization technique, we propose in this paper a reconfigurable FFT (Fast Fourier Transform) operator. This operator can be reconfigured to switch from an operator dedicated to compute the FFT in the complex field (i.e for OFDM modulation or frequential equalization) to an operator which computes the FFT in the Galois Field in order to perform Reed-Solomon (RS) encoding and two steps of the decoding process.

I. INTRODUCTION

The "SoftWare Radio" concept was first introduced in the literature around 1990 thanks to the pioneering works of J. Mitola [1] and W. Tuttlebee [2]. SWR basically refers to an ensemble of techniques which permits the reconfiguration of a communication system without the need to change any hardware system element. This reconfiguration implies the optimization of the hardware-software resources in the terminal architecture design. So as to help this optimization, a new area of research called "parametrization" has appeared, whose goal is to identify common resources, i.e Common Operator (CO) or Common Function (CF) between all the standards involved in the reconfiguration and in the standards themselves. In [3], the CO approach is presented and [3] constitutes the first paper in which the parametrization is defined. In [4] the same authors, proposed the FFT as a common operator and show how it can make a basic function in many classical telecommunications operations, turning the algorithms into the frequency domain [5]. Until now, this operator was never used in the channel decoding function. In this paper we will show that the FFT regarded as a reconfigurable common operator could be used in some operations for RS encoding and decoding to reduce the run-time execution and the complexity of RS decoders. The paper is organized as follows: section II presents a brief summary of the state-of-the-art involving RS decoders architecture in the frequency and time domains. In section III we will identify the specific RS code defined over the Galois Field $GF(F_n)$, where F_n is the Fermat number. The encoding and two steps of the decoding process of this code

can be performed with the FFT reconfigured in such way to be considered as the Fermat Number Transform. In section IV we present the reconfigurable FFT operator and the new frequency architecture for RS encoder-decoder to be defined. Finally, the conclusions are outlined in section V.

II. GENERAL PRINCIPLES OF FREQUENTIAL REED SOLOMON CODES

Reed-Solomon codes are considered as ones of the most powerful algebraic codes and have found many applications in telecommunications in the last years. These codes which occupy a prominent place in the theory and practice of error correction have a certain optimality property and a well-understood distance structure. The following is a list of many applications that use the RS codes: ADSL, VDSL, HDSL, SDSL, CDplayers, DVD, DVB-T, DTV, ATSC, Mobile systems geo-synchronous satellite communications links, CD-ROMs, Wireless communications, Deep-space probe missions, Interplanetary reconnaissance,... . In the following subsections, after having defined the Fourier transform over a finite field, we will briefly describe the encoding and decoding in the frequency and time domains for RS codes.

A. The Fourier Transform

Galois Field $GF(q)$ is a finite set composed of q elements allowing an algebraic, methodical treatment of error correcting codes. Each GF has a primitive element α meaning that any other element of GF can be expressed as a power of α .

In the complex field (\mathbb{C}), the Discrete Fourier Transform of $v = (v_0, v_1, \dots, v_{N-1})$, a vector of real or complex numbers, is a vector $V = (V_0, V_1, \dots, V_{N-1})$, given by

$$V_k = \sum_{i=0}^{N-1} e^{-j \frac{2\pi i k}{N}} v_i \quad k = 0, \dots, N-1 \quad (1)$$

where $j = \sqrt{-1}$. The Fourier kernel $\exp(-j2\pi/N)$ is an N th root of unity in the field \mathbb{C} . In the finite field $GF(q)$, an element α of order N is an N th root of unity. Drawing on the analogy between $\exp(-j2\pi/N)$ and α , we have the following definitions:

Let $v = (v_0, v_1, \dots, v_{N-1})$ be a vector over $GF(q)$, and let α be an element of $GF(q)$ of order N . The vector v and its Discrete Fourier Transform are related by :

$$V_j = \sum_{i=0}^{N-1} \alpha^{ij} v_i \iff v_i = \frac{1}{N} \sum_{j=0}^{N-1} \alpha^{-ij} V_j, \quad (2)$$

for $j = 0, \dots, N-1$, where N is interpreted as an integer of the field. Further details can be found in [6].

B. RS encoding based on the Fourier transform

Applications of the Discrete Fourier Transform in the complex field occur throughout the subject of signal processing. Fourier transforms also exist in the Galois Field $GF(q)$ and can play an important role in the study and processing of $GF(q)$ valued signals, that is, of codewords. By using the Fourier transform, the ideas of coding theory can be described in a setting that is much closer to the methods of signal processing. In frequency-domain, cyclic codes can be defined as codes whose codewords have certain specified spectral components equal to zero [6]. Thus, one can define a RS code of block length N over $GF(q)$, with N a divisor of $q-1$, and minimum distance d as follows. Given a set of spectral indices, $\mathbf{B} = \{j_0, j_0 + 1, \dots, j_0 + d - 2\}$, whose elements are called *check frequencies*, the RS code is the set of words over $GF(q)$ whose spectrum is zero in components indexed by $j_0, j_0 + 1, \dots, j_0 + d - 2$. One can choose any j_0 for a RS code. Then the generator polynomial is given by

$$g(x) = (x - \alpha^{j_0})(x - \alpha^{j_0+1}) \dots (x - \alpha^{j_0+d-2}).$$

One may encode in the natural way using a generator polynomial. We call this encoder a *time-domain encoder*. Alternatively, one may choose to encode the RS code directly in the transform domain by using the data symbols to specify spectral components. We call this a *frequency-domain encoder*. Encoding is as follows. Some set of $d-1$ cyclically consecutive frequencies, indexed by $j = j_0, j_0 + 1, \dots, j_0 + d - 2$, is chosen as the set of spectral components constrained to zero. The $N-d+1$ unconstrained components of the spectrum are filled with data symbols $GF(q)$. The inverse Fourier transform then produces a nonsystematic codeword.

C. RS decoding based on the Fourier Transform

Usually, there are two principal ways to decode a given RS code. The first one, stated in the language of spectral estimation, consists of a Fourier transform (syndrome computation), followed by a spectral analysis (Berlekamp-Massey algorithm), followed by an inverse Fourier transform (Chien search). The second consists to push the Berlekamp-Massey algorithm into the time-domain to work directly on the raw data word as received without the usual syndrome calculation or power-sum-symmetric functions [7].

In [8] two architectures for universal time-domain RS decoders are given. The first universal decoder has a very simple structure and takes N^2 clocks to decode one codeword, where N is the blocklength of the code. The decoding time does not depend on the number of errors or erasures in the received word. The second universal decoder has a more complex structure but is faster. It takes $2tN$ clock intervals to decode one

codeword, where t is the code correction power. On the other hand, the frequency-domain vectors of length t are replaced by time-domain vectors of length N ; then these decoders which have N^2 and Nt complexity become decoders in the frequency domain with Nt and t^2 complexity respectively. The time-domain decoder is structurally simple but the penalty is a longer running time. In [9] a versatile Reed-Solomon decoder has been developed based on the time-domain decoding algorithm. The decoding time of this decoder is $N(N+1)T$ where T represents the clock cycle that is determined by the longest delay path. In [10] a pipeline frequency-domain RS decoder is developed in an ATM network context. The decoding process of this pipeline decoder requires $2N+10t+3$ clock cycles where the decoding stages could be gathered in three phases as shown in Figure 1.

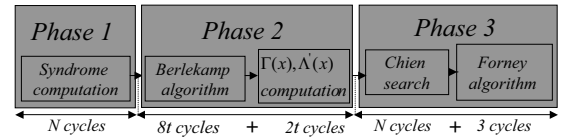


Fig. 1. The three phases of RS decoding process

Phase 1 and 3 have the longer execution time and have a similar time duration. We did not take into account the duration of erasure computation because it can be regarded as part of the Berlekamp-Massey (initialization). In the next section we will explain that the clock cycles of both phases "1" and "3" can be decreased by using the FFT operator in the case of RS codes defined over $GF(F_n)$.

III. RS CODES OVER $GF(F_n)$

The frequency-domain decoders described previously are based on the Fourier transform over $GF(q)$ where $q = 2^n$ and 2 is the primitive element of order N , where $N = 2^n - 1$ represents the codeword length. The first disadvantage of the transform over $GF(2^n)$ is that its length is an odd number, so that the most efficient FFT algorithm cannot be used to yield a fast transform decoder. The second disadvantage is that the arithmetic required to perform these transforms over $GF(2^n)$ still requires a substantial number of multiplications in $GF(2^n)$.

Rader [11] proposed transforms over rings of integers modulo both Mersenne and Fermat numbers that can be used to compute error-free convolutions of real integer sequences. Agarwal and Burrus [12] extended Rader's Fermat number-theoretic transform by using the generator $\alpha = \sqrt{2}$ for the transform rather than $\alpha = 2$. In this case the usual FFT algorithm can be used to calculate transforms with as many as 2^{n+2} points of integer data. Justesen [13] proposed that transforms over the finite field $GF(F_n)$, where $F_n = 2^{2^n} + 1$, of integers modulo a Fermat prime can be used to define RS codes and to improve the decoding efficiency of these codes. It is known that $\sqrt{2} \in GF(F_n)$ for $n=2,3,4$ [14] is an element of order 2^{n+2} in $GF(F_n)$. Consequently, a number-theoretic transform decoder for RS that uses $\sqrt{2}$ as a root of unity can

handle as many as 2^{n+2} symbols for $n=2,3,4$. In order to treat longer RS codes and to make use of fast Fourier transform, it is necessary to find roots of unity in $GF(F_n)$ of higher order [13]. In particular, let us consider roots of order 2^{n+4} .

Since $2^{2^n} \equiv -1 \pmod{F_n}$, $2^{(F_n-1)/8} \equiv 2^{2^n \cdot 2^{n-3-n}} \equiv (-1)^{2^{2^n-3-n}} \equiv 1 \pmod{F_n}$ for all $n \geq 3$. As a consequence $\sqrt[8]{2}$ is an element of $GF(F_n)$. Since $(\sqrt[8]{2})^{2^{n+3}} \equiv -1 \pmod{F_n}$, $\alpha = \sqrt[8]{2}$ is an element of order 2^{n+4} in $GF(F_n)$. Thus the FFT over $GF(F_n)$ can be defined to compute the transform of a sequence of as many as $N=2^{n+4}$ points of integer data. Similarly, one can demonstrate that $\alpha = \sqrt[16]{2}$ is an element of order 2^{n+5} for $n=3$ what allows to obtain a maximum length transform and then define Reed-Solomon code with blocklength equal to $F_n-1=256$. In [14] a decoding algorithm for RS code using FFT over $GF(F_n)$ is given. The described algorithm is the direct approach adopting the recursive extension to evaluate the error magnitudes. We will show now how the FFT can be used to encode in the frequency domain and involved in two stages of the decoding process which uses the Forney algorithm to evaluate error magnitudes (algorithm shown in Fig. 1).

1) *Frequency-domain encoding over $GF(F_n)$* : As we mentioned in section II, encoding in frequency-domain consists to set at zero a specified block of spectral components and to fill the unconstrained components of the spectrum with data symbols. Then, to obtain a code word, the inverse Fourier transform can use a fast transform composed of $\log(F_n - 1)$ stages.

2) *Frequency-domain decoding over $GF(F_n)$* : The decoding of RS codes constructed over $GF(F_n)$ using a FFT and the algorithm described in Fig. 1, is composed of three main phases including the following steps :

Phase 1 :

◊ Compute the FFT over $GF(F_n)$ of the received code N -tuple r_i ,

$$S_k = \sum_{i=0}^{N-1} \alpha^{ik} r_i \quad (3)$$

where the components $S_{j_0} \dots S_{j_0+2t}$ represent the syndromes and α is an element of order N .

Phase 2 :

(a) Use Berlekamp-Massey algorithm [6] to determine the error-locator polynomial Λ_i from the known S_j .

(b) Compute the error evaluator polynomial $\Gamma(x)$ from the defining equation

$$\Gamma(x) = \Lambda(x)S(x) \pmod{x^{2t+1}}$$

and the derivative of $\Lambda(x)$

$$\Lambda'(x) = \Lambda_1 + 2\Lambda_2x + \dots + t\Lambda_t x^{t-1}.$$

Phase 3:

(a) Compute the FFT of vector $(\Lambda_0, \Lambda_1, \dots, \Lambda_t, 0, \dots, 0)$ to perform Chien search. The spectral components which are

equal to zero indicate the roots of $\Lambda(x)$. Indeed, "Chien search" consists to test the sum $1 + \Lambda_1\alpha^l + \Lambda_2\alpha^{2l} + \dots + \Lambda_t\alpha^{tl}$; if this sum is zero, then α^l is a root of $\Lambda(x)$. In addition, the l th spectral component of $\text{FFT}(\Lambda(x)) = \sum_{i=0}^{N-1} \Lambda_i\alpha^{il}$ is nothing else than the value of $\Lambda(x)$ at α^l . If this spectral component is equal to zero then α^l is a root of $\Lambda(x)$ and the received symbol r_{n-l} is erroneous.

(b) Apply the Forney algorithm, related to the following equations :

$$c_i = r_i + \frac{\alpha^{-i}\Gamma(\alpha^{-i})}{\Lambda'(\alpha^{-i})} \text{ if } \Lambda(\alpha^{-i}) = 0$$

$$c_i = r_i \text{ if } \Lambda(\alpha^{-i}) \neq 0$$

This is a way to demonstrate that the FFT operator can perform efficiently the syndrome computation and Chien search decreasing their time-computation from N to $\log(N)$ clock cycles as we will see in the next section.

IV. NEW FREQUENCY ARCHITECTURE

In the previous section, we have identified that the FFT can be used in the RS encoding and decoding processes over $GF(F_n)$. We will now present the operator structure and the advantages it can present.

The "Butterfly" operation specifically adapted to the FFT algorithm over \mathbb{C} can be applied too over $GF(F_n)$ where the primitive element α replaces the Fourier kernel $\exp(-j2\pi/N)$ (Fig.2).

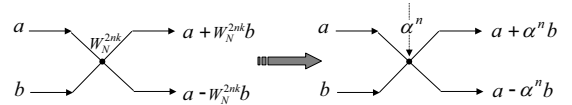


Fig. 2. The FFT Butterfly over \mathbb{C} and $GF(F_n)$

If $\alpha = \sqrt[m]{2} \in GF(F_n)$ is an element of order $2^{n+\log(m)+1}$, the FFT algorithm over $GF(F_n)$ has $l = n + \log(m) + 1 = \log(N)$ stages of computation, where N is the vector length. Consequently, the steps of "N cycles" of syndrome computation in phase 1 and the execution of Chien search in phase 3 are reduced to " $\log(N)$ cycles" steps. In order to perform transforms over $GF(F_n)$, one needs to find an element of order N . To achieve this, an IBM assembler language program was used to compute the solution of $x^{\frac{N}{2}} \equiv -1 \pmod{F_n}$ [13].

In Figure 3 we give an example of a diagram for the $N=16$ -point radix-2 decimation in time FFT over $GF(F_2 = 17)$.

This "Butterfly-structure" which represents $l=4$ stages of computation, is based on the following characteristics of (3):

- a) $\alpha^{k+N} = \alpha^k$, since $\alpha^N = 1$
- b) $\alpha^{k+\frac{N}{2}} = -\alpha^k$.

It should be pointed out that a word length of $2^n + 1$ bits is required to represent a number in $GF(F_n)$ and the arithmetic operations such as addition, subtraction, multiplication and division are operations modulo F_n .

As previously mentioned, the main goal of our study is to identify the FFT as a reconfigurable operator. In the context

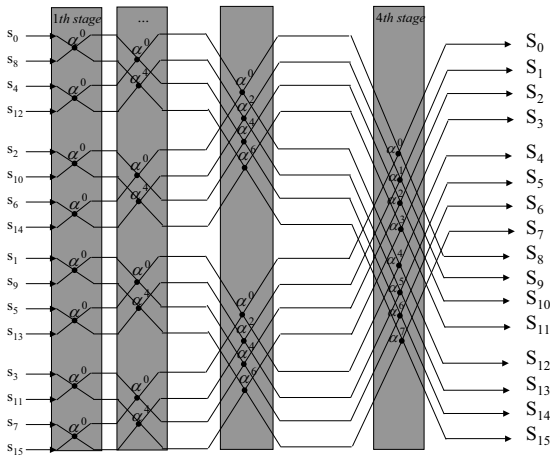


Fig. 3. The diagram of 16-point radix-2 decimation in time over $GF(F_n)$

of parametrization, the FFT operator can be reconfigured to switch from an operator dedicated to compute the Fast Fourier Transform in the field of complex numbers to an operator that will perform the two most long-time stages for the RS decoding. In this context we will define the new FFT operator as shown in Figure 4. One can consider two operating modes: the first one is the Fast Fourier Transform computation over \mathbb{C} ; then the Fourier kernel $exp(-j2\pi/N)$ is downloaded and the block "Mod F_n " is switched to an idle mode. The second one is the Fast Fourier Transform computation over $GF(F_n)$; in this operating mode, the primitive element α^n is downloaded and the block "mod F_n " is switched on to perform the division modulo F_n for the output of the "FFT" block. Figure 5 represents the global frequency-domain architecture

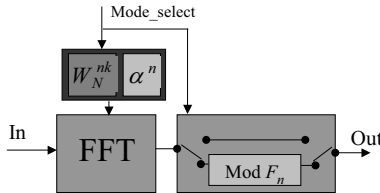


Fig. 4. The reconfigurable FFT operator

of encoder/decoder for Reed-Solomon codes. The decoding algorithm presented in this Figure is an improvement of the one presented in Figure 1 with some changes. Now, The RS code is constructed over $GF(F_n)$ rather than $GF(2^n)$. One notes that this specific code is recommended for Spacecraft communication [15] where the RS(256,224) over $GF(257)$ is used. In this case the FFT can operate efficiently in both encoding and decoding processes. Then, the encoder uses the data symbols in the frequency domain and the decoder does not compute an inverse Fourier transform. The corrected spectrum gives the data symbols directly (see Figure 5).

V. CONCLUSION

In this paper we have proposed a reconfigurable operator for frequential computations applied both to RS encoding and

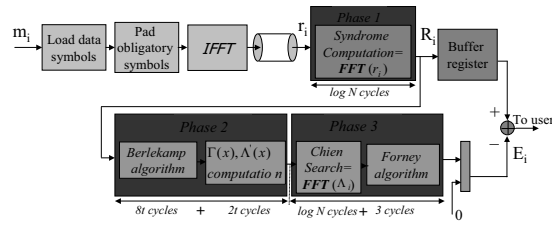


Fig. 5. The frequency-domain encoder/decoder for RS codes over $GF(F_n)$

decoding in $GF(F_n)$ and to the complex numbers field transforms. This operator dedicated to operate over $GF(F_n)$ decreases the complexity of syndrome computation and "Chien serach" to $log(N)$ cycles, compared to N cycles for RS codes constructed over $GF(2^n)$. Then we have shown that the structure of this operator is the same over $GF(F_n)$ and \mathbb{C} fields and is reconfigurable, depending on the downloaded kernel of the performed FFT mode. As a consequence, this operator can be considered as "Common" in a parametrization context where classical frequential operation (i.e OFDM or frequential equalization) and RS channel decoding are involved in a reconfigurable scenario. Future work should focus first on the practical realization of this reconfigurable operator and second on its lower level of reconfigurability.

REFERENCES

- [1] J. Mitola, "The software Radio Architecture", IEEE Communications Magazine, May 95, pp. 26-38.
- [2] W. Tuttlebee, "Evolution of radio systems into the 21st century", Proc.IEEInt. Conf. on 'Radio receivers an associated systems',1995.
- [3] J. Palicot, D. Giri, C Moy, "A Theoretical Approach of Parameterization Design for SDR Systems", Workshop on Software Defined Radio : theory, design and applications, ESSIRC 2005 - Grenoble -France
- [4] J. Palicot, C. Roland, "FFT: a basic Function fo a Reconfigurable Receiver", ICT' 2003, Feb. 2003, Papeete, Tahiti.
- [5] K. Berberidis, S. Rantos and J. Palicot, "A Step-by-Step Quasi-Newton Algorithm in the Frequency Domain and its Application to Adaptive Channel Equalization", IEEE Transactions on Signal Processing, Vol.52, No.12, Dec. 2004, pp.3335-3344
- [6] Richard E. Blahut, "Algebraic Codes for Data Transmission", Cambridge University press, 2003.
- [7] R. E. Blahut, "Transform Decoding without Transform", presented at the Tenth IEEE Communication Theory Workshop, Cypress Gardens, FL, 1980.
- [8] R. E. Blahut, "A universal Reed-Solomon decoder", IEEE. Trans. Comput., vol.C-34,pp. 150-158. Mar. 1984.
- [9] Youssef R. Shayan, Tho Le-Ngoc and Vijay K. Bhargava. "A versatile Time-Domain Reed-Solomon Decoder", IEEE Journal on Selected Areas in Communications", Oct. 1990.
- [10] Antonio Gabriel Lomeña, Juan Carlos López and Ander Royo. "A Pipeline Frequency-Domain Reed-Solomon Decoder for Application in ATM Networks", XIV Design of Circuits and Integrated Systems Conference, Palma de Mallorca, 16-19 Nov. 1999.
- [11] C. M. Rader, "Discrete convolutional via Mersenne transforms", IEEE Trans. Comput., vol. C-21, no. 12, pp. 1269-1273, Dec. 1972.
- [12] R. C. Argawal and C. S. Burrus, "Number theoretic transforms to implement fast digital convolutional", Proc. IEEE, vol. 63, no. 4, pp. 550-560, Apr. 1975.
- [13] J. Justesen, "On the complexity of decoding Reed-Solomon codes", IEEE Trans. Inform. Theory, vol. IT-22, pp. 237-238, Mar. 1976.
- [14] Irving S.Reed, T.K. Truong, and L. R. Welch, "The Fast Decoding of Reed-Solomon Codes Using Fermat Transforms", IEEE Transactions on Information Theory, vol. IT-24, No. 4, Jul. 1978.
- [15] BEST M R; ROEFS H F A, "Technical assistance channel coding investigation (spacecraft Telemetry)" [Final Report] 1981.